Multivariate Time Series and Machine Learning Techniques for the multi-class Classification of Lathe's Cutting Tools Wear Condition

Thiago E. Fernandes * **Matheus A. M. Ferreira** *
Guilherme P. C. de Miranda * **Marcos V. G. R. da Silva** *
Eduardo P. de Aguiar *

* *Federal University de Juiz de Fora, Juiz de Fora, MG, (e-mail: {thiago.esterci, matheus.malta, guilherme.pires, marcos.victor, eduardo.aguiar}@engenharia.ufjf.br).*

Abstract:
The majority of mechanical components went through a machining process during their manufacturing. Therefore, manufacturing processes with inadequate condition tools are likely to induce unexpected operational interruptions, accidents, product quality, and economic losses. Accordingly, the ability to classify fault imminences can result in cost reduction, along with productivity and safety increase. This paper aims to discuss an autonomous model based on the Self-Organised Direction Aware Data Partitioning Algorithm (SODA) and machine learning techniques, including time series Feature Extraction based on Scalable Hypothesis tests (TSFRESH), to solve this problem. The model proposed in this work can identify the patterns that distinguish the cutting tool's flank wear in a multi-class scenario as adequate, intermediate, and inadequate conditions, achieving satisfactory performances in all cases and allowing to prevent fault occurrences.

Keywords:
Autonomous Learning; Empirical Data Analyses; Machine Learning; Machining Processes.

1. INTRODUCTION

The 2020 World Machine Tool Survey (Gardner Business Media, 2021) shows a downturn regarding global machine tool consumption and production comparing with previous years. The coronavirus and resulting economic lockdowns in numerous countries had a significant impact on the machine tool market.

Global machine tool consumption in 2020 was US$66.8 billion, down 20.1% from 2019. According to (Steven Kline, 2021), taking into consideration the consumption decline during 2008-2009 financial crisis, the effects of the economic lockdowns on the global machine tool market are not as severe as could have been expected.

Nevertheless, markets of primary and intermediate goods, such as automotive or electronic components and durables goods, tend to recover. Considering that the demand for machine tools comes from these manufacturers, a similar growth of machine tools consumption is expected for the following years. In this scenario of financial recovery, the increasing competitiveness leads the companies to demand reduction in tools consumption, maintenance time, and improvements in manufacturing processes quality, availability and reliability.

The innovation in smarter maintenance procedures is likely to results in the replacement of the schedule-based maintenance by condition-based maintenance (Lee et al., 2010). Therefore, incorporating machine learning techniques, particularly time series analyses, into smarter maintenance operations may increase machining process reliability while also reducing machine down time, resulting in lower maintenance costs. In this context, our work is essential, since it integrates data monitoring and machine learning to the machining process.

Time series are used in various fields of application to understand and analyze the evolution of a phenomenon over time. If the observed variables are more than one, the series is called multiple or multivariate. The time series used in this work are a collection of data acquired at regular intervals from current and voltage sensors from the lathe's spindle motor.

Selecting pertinent and representative features from data is one of the major challenges when analyzing time series. To overcome this challenge, a methodology named TSFRESH was adopted in this work Christ et al. (2018). The TSFRESH algorithm extracts 74 different features from the time series. Additionally, it applies feature selection by hypotheses test. The hypotheses test method applied in the feature selection is the Kolmogorov-Smirnov (KS) (Wilcox, 2005). The KS test is based on calculating the highest difference between cumulative distributions between two random variables. Hence, the KS test is designed for binary classification problems. Although the TSFRESH method was adopted in many models included in the faulty detection literature, the implementation of this method in a multi-class scenario holds an unexplored prospect. In this work, we propose an enhanced TSFRESH method for multi-class feature selection.

Aiming to minimize user interference in the model, the Self-Organised Direction Aware Data Partitioning Algorithm (SODA)(Gu et al., 2018) was adopted in this work. The SODA can self-adjust the data cloud structure and parameters to follow possible changes in data patterns and processes. Moreover, the SODA algorithm considers both spatial and angular divergence, resulting in a more accurate similarity recognition among the data than traditional clustering/partitioning methods. The application of this algorithm in different engineering problems (Fernandes and de Aguiar, 2021; Pinto et al., 2022) vogue for SODA's capability to adapt to various types of data.

In this context, the main contributions of this work are summarized as follows:

- We propose, for the first time in the literature, a method for multiclass feature selection based on Scalable Hypothesis tests. This method made it possible to study the tool's wear state prior to its failure, enabling the model to prevent inadequate condition operations.
- We present and evaluate two different solutions for applying the multiclass feature selection on a data set recorded by a data acquisition system developed in the UFJF's the Laboratory of Industrial Automation and Computational Intelligence (LAIIC).
- We propose an autonomous approach to classifying lathe cutting tools in a multi-class scenario (as an adequate, intermediate, and inadequate condition), does not require prior expert knowledge, and improves the machining process reliability.

And our major conclusions are:

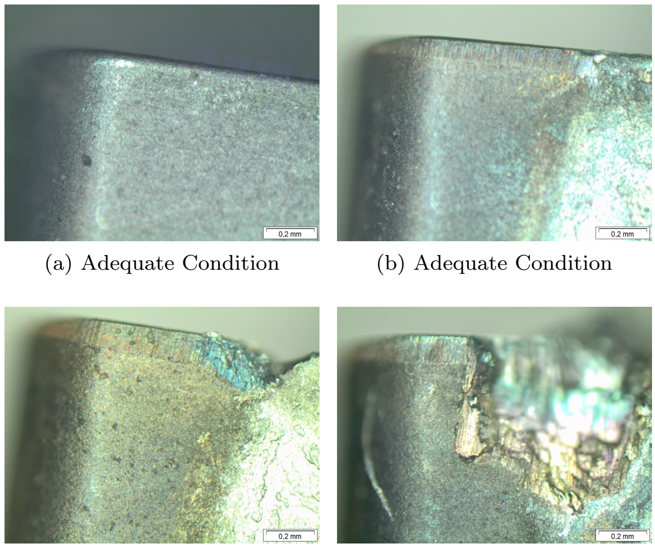
- The proposed approach can identify the patterns that distinguish the lathe's cutting tool between adequate, intermediate and inadequate, achieving satisfactory performances in all cases, and enabling to prevent faulty pieces fabrication.
- Considering achieved results, the multi-class Time Series Feature Extraction on basis of Scalable Hypothesis tests (TSFRESH) (Christ et al., 2018) is entirely suitable for voltage and current time series from a lathes' three-phasic spindle motor, specially with the implementation of the proposed multi-class KS test.
- The development of a lathe's cutting tool diagnosis methodology based on the TSFRESH, SODA, and Machine Learning Techniques is reasonable due to the positive aspects in analyzing voltage and current time series from a lathes' three-phasic spindle motor, such as the ability to manage uncertainties. The numerical examples in this paper demonstrate that the proposed autonomous model produces high-quality clustering results.

This paper is organized as follows: Section 2 states the problem formulation. Section 3 discusses the data acquisition system. Section 4 discusses the methods adopted in the proposed model. After that, Section 5 discusses the numerical results. Finally, the Section 6 closes the work and exhibits the conclusions with respect to the stated propositions.

2. PROBLEM FORMULATION

The replacement of the cutting tool earlier or later than necessary will cause either loss of resources or damaging products. However, the replacement of the cutting tool remains supported by schedule-based maintenance. The cutting force is essentially a function of the cutting speed, the feed rate, and the depth of the cut. Consequently, the tool replacement schedules depend on these same cutting parameters (Li et al., 2000). Note that the tools are replaced without further inspection. Since machining is a highly dynamic process, this method can lead to greater machinery downtime, waste of good condition tools, and fault occurrences.

This work focus on the monitoring and diagnoses of the cutting tool's flank wears. The development of the flank wear takes place in the contact area between the cutting tool and the work-piece. This contact area causes the erosion of the cutting tool by friction. Figure 1 presents the flank wear evolution on a cutting tool studied in this work.



(a) Adequate Condition (b) Adequate Condition

(c) Intermediate Condition (d) Inadequate Condition

Figure 1. Lathe's cutting tool flank wear evolution

The evolution of the flank wear causes an increase in the contact area at the tool-work-piece interface. Consequently, friction and machining forces are intensified, which results in greater power consumption by the process. Therefore, more energy is consumed when machining with a worn tool than with a new tool (Shao et al., 2004). The electric voltage and current of the lathe's motor, have been broadly used in cutting tools' wear monitoring. The monitoring of other physical quantities such as acoustic emission, forces, vibration, and temperature require sensors in moving parts and near the tool-work-piece interface. The electric voltage and current were chosen due to their simple implementation in machining systems. It is worth mentioning that no other physical quantity has been investigated in this work.

Therefore, the time series used in this work is a collection voltage and current of the three phases of the lathes' motor. The time series were recorded from operations in a real machining system at the National Industrial Training Service (SENAI) in collaboration with the Laboratory of Industrial Automation and Computational Intelligence (LAIIC). This procedure used in this work is regulated by ISO 3685/1993. It consisted of executing successive machining operations and examining the regular intervals of the tool wear condition. This process is repeated until the tool wear reaches a pre-established limit. Considering the technical guidelines of ISO 3685/1993, the adopted limit was the flank wear's maximum length of 0.6 mm. Thus, the flank wears greater than the established limit was considered an inadequate condition. Furthermore, the machining conditions were: depth of cut of 0.5 mm, cutting speed of 120 m/min, and feed rate of 0.156 mm/rev.

The machining process was executed in a 2014 Romi GL280M Turning Center with FANUC'S CNC exhibited in Figure 2. This machine was equipped with a set of interchangeable TNUX 160- 404 R LT 1000 inserts. The time series was acquired during the experiments using a data acquisition system developed at the Laboratory of Industrial Automation and Computational Intelligence (LAIIC).

Figure 2. 2014 Romi GL280M Turning Center with FANUC'S CNC.

3. DATA ACQUISITION SYSTEM

The data acquisition system developed at LAIIC consists of an embedded electronic board that read sensors information, convert the acquired data into digital form and store the information onto a SD Card.

The analog sensors used in this work are three non invasive current sensors SCT-013 and three voltage sensors ZMPT101B-250V, connected to the lathe's three phase spindle motor. The current sensors operate acquiring signals in the range of $\pm 100A$ and the voltage sensor in the range of $\pm 250V$. Afterwards the sensor converts the signal to an output voltage between 0 to 5V.

The acquisition module requires a 16-bit resolution in order to achieve the necessary accuracy in the measures. Therefore the Analog Devices Inc. Analog/Digital Converter AD7606 is used in the acquisition system.

The analog signal, from the sensor, is sent to the AD7606. The main advantage of AD7606 is the simultaneous acquisition system of its eight channels that can lead to 200 KS/s. After the conversion, the digital signal are sent to the STM32F407ZGT6.

The STM32F407ZGT6 is based on the high-performance Arm Cortex-M4 32-bit RISC core operating at a frequency of up to 168 MHz, with a 1 MB Flash memory and 192 Kb SRAM. On Figure 3 a Finite State Machine (FSM) explains the STM32F407ZGT6 algorithm.

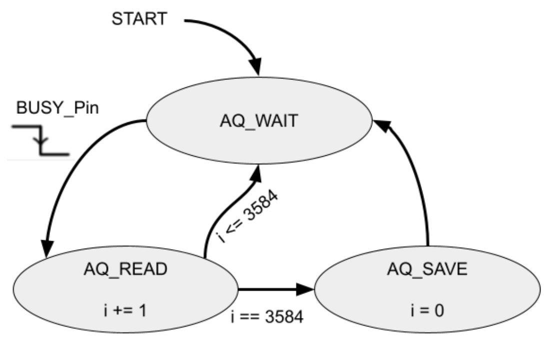


Figure 3. STM32F407ZGT6 algorithm diagram.

The states of the FSM can be described as:

- **AQ_WAIT**: Wait for the falling edge of $BUSY\_Pin$, end of AD7606 conversion;
- **AQ_READ**: Read and store into memory one sample of each AD7606 channel. Increment i;
- **AQ_SAVE**: When $i == 3584$ (time series length) save time series onto the SD Card and reset i value.

4. LATHES TOOL MODEL

The structure of the model is presented in Figure 4. The data set **R** consists of 785 time series recorded with the prototype developed in this work. Each time series has 3584 measurements of voltage and current of the lathe's three-phase spindle motor.

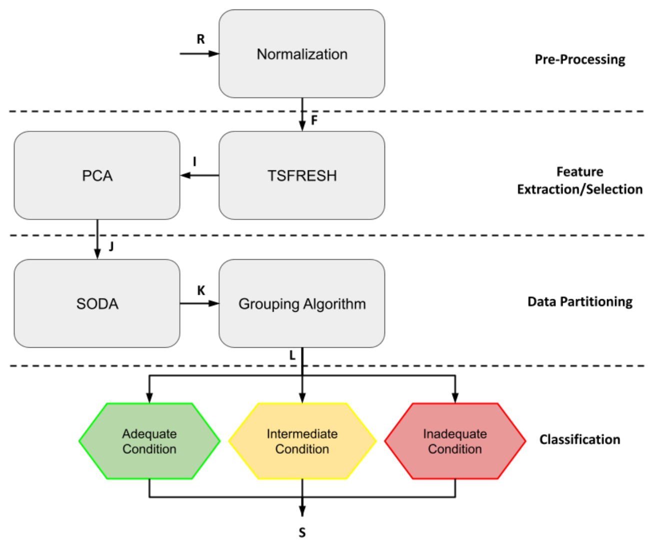


Figure 4. Proposed Model

After the pre-processing stage, the matrix **F** is presented to the TSFRESH algorithm, which is responsible for

ISSN: 2525-8311

DOI: 10.20906/CBA2022/3585

extracting 74 different features of each variable of the time series, as well as selecting, through hypothesis test, which of those are relevant for characterizing the fault occurrences. The output of TSFRESH consists of a matrix **I**.

Aiming to reduce the **I** matrix dimensionality, the Principal Component Analysis (PCA) method was applied. The PCA output, represented by **J**, results from the projection of the data of matrix **I** in a new coordinate system formed by axes, named Principal Components (PC's), which are calculated at the beginning of this process. Note that, after this projection, the data in matrix **J** can not be directly related to a specific sensor nor with a specific feature from the previous process, since the PC's are calculated from the variation of the data within all dimensions of matrix **I**. On the other and, the relevance of each feature and sensor, in this process, will be further discussed in Section **??**.

In order to group the data in matrix **J** into Data Clouds, the Data partitioning Module commences by presenting the matrix **J** to the SODA algorithm. The data clouds are formed by calculating distance and dissimilarity metrics among the data points. Consequently, the SODA's output, represented by **K**, consists of the same data presented in matrix **J**, however, with labels defining which data cloud the data points belong.

Afterwards, the Data Partitioning Module is concluded by the Grouping Algorithm, which gathers the data clouds into the two groups that follow: adequate condition tool and inadequate condition tool. Therefore, the Grouping Algorithm's output, represented by **L**, is very similar to the matrix **K**, though each data point bears a new label granted by this algorithm.

Finally, the data is presented to the Classification stage, in which the labeled matrix **L** is used to train different classifiers in the task of classifying a cutting tool condition as adequate, fault imminence and inadequate. Note that all steps, exposed above, will be discussed further in Sections 5.1.

4.1 Normalization

Let $\mathbf{F}$ be a set of time series such that $\mathbf{F} = \{\mathbf{t}_1, \mathbf{t}_2, ..., \mathbf{t}_j\}$ with $j = 785$ where $\mathbf{t}_k$ represents the k^{th} time series of the normalized values of voltage and current from $\mathbf{T}_k$. The normalization applied follows the equation 1.

$$\mathbf{t}_k(i,j) = \frac{\mathbf{T}_k(i,j) - \mu_i}{\sigma_i} \tag{1}$$

Where $\mathbf{t}_k(i,j)$ is the normalized data of the i^{th} variable at the j^{th} measurement, μ_i is the mean of the time series and σ_i is the standard deviation of the corresponding time series variable. Note that each variable of the time series is normalized independently.

4.2 Feature Extraction on basis of Scalable Hypothesis tests (TSFRESH)

The TSFRESH framework (Christ et al., 2018) is capable of extracting 77 features from each variable of the time series. These features are extracted with different parameters. Hence, one feature returns multiple outputs. For instance, the Fourier Transformation returns the real part, the imaginary part, the absolute value, and the angle in degrees.

Considering X a feature calculated for one of the variables within $\mathbf{t}_k$, the k^{th} time series from $\mathbf{F}$, the relevance of X (Christ et al., 2018) to a target Y is calculated as the difference between their conditional distribution and expressed as $f_{X|Y=y_1}$ and $f_{X|Y=y_2}$, where y_1 and y_2 are the set of values for feature X calculated for the time series of adequate and inadequate condition tools respectively. Therefore, feature X is relevant to estimate Y if, and only if:

$$\exists y_1, y_2 \;\; with \;\; f_y(y_1), f_y(y_2) > 0 : f_{x|y=y_1} \neq f_{x|y=y_2} \tag{2}$$

Equation 2 also corresponds to X and Y being statistically dependents. Feature X is irrelevant when:

$$\exists y_1, y_2 \;\; with \;\; f_y(y_1), f_y(y_2) > 0 : f_{x|y=y_1} = f_{x|y=y_2} \tag{3}$$

and it also means that X and Y are statistically independents.

The relevancy can also be investigated through hypothesis test christ2016distributed. To the extracted features $X_1, X_2, ..., X_n$, a hypothesis test is applied independently, in order to investigate the following hypothesis:

$$\begin{aligned} H_0^i &= X_i \text{ is not relevant to } Y \\ and \;\; H_1^i &= X_i \text{ is relevant to } Y \end{aligned} \tag{4}$$

The result of each test is called p-value and corresponds to the probability of obtaining a measure of equality or inequality between the hypothesis test and the observed in the data based on the null hypothesis. In this work, the p-value measures if the analyzed feature is relevant or not and small p-values show more relevant features.

The test applied in this paper is the Kolmogorov-Smirnov (KS) wilcox2005kolmogorov, considering the following hypotheses:

$$\begin{aligned} H_0^i &= \left\{ f_{X_i|Y=y_1} = f_{X_i|Y=y_2} \right\} \\ H_1^i &= \left\{ f_{X_i|Y=y_1} \neq f_{X_i|Y=y_2} \right\} \end{aligned} \tag{5}$$

where $f_{X_i|Y=y_1}$ is the cumulative distribution function (CDF) of feature X considering one class and $f_{X_i|Y=y_2}$ is the CDF of feature X_i considering another class.

The KS test considers the maximum difference between the CDF obtained from the features, a shown in Equation (6).

$$D = \sup |f_{X_i|Y=y_1} - f_{X_i|Y=y_2}|. \tag{6}$$

It is worth emphasizing that the KS test identifies features that can distinguish between two classes. However, this work deals with a multi-class problem. Aiming to provide features capable of distinguishing all desired classes, the strategy presented in Figure 5 was applied.

Firstly, the feature extraction by TSFRESH is applied to the data set. The extracted features are separated by

ISSN: 2525-8311

DOI: 10.20906/CBA2022/3585

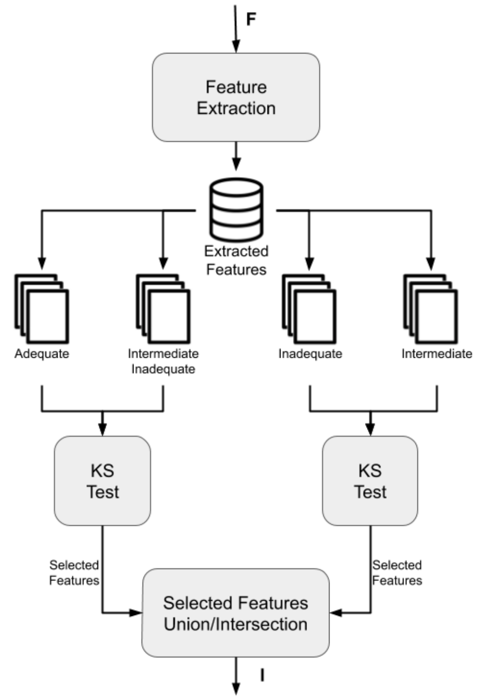


Figure 5. TSFRESH detailed description.

tool condition in order to execute two separated KS tests. The first test aims to selected features to distinguish the adequate from the intermediate and inadequate condition tools. Afterward, the KS test is applied to select features that can distinguish intermediate from inadequate conditions tools. Finally, the proposed algorithm computes the intersection or union between the selected features of these two hypothesis tests. The output of TSFRESH consists of a matrix **I**. It is worth mentioning that the comparison between these two approaches (intersection or union) is presented in Section 5.1.

4.3 Principal Component Analysis

Given a set of variables $X = x_i$, with $i = 1, 2, 3, ..., n$, it is possible to investigate a smaller set of variables, within X, in which their linear combination $\alpha_{\mathbf{k}}\mathbf{X}$ preserves a major part of the information available in X with maximum variance. These variable are called principal components and the first of the principal components is z_i, known for comprising the major variability of the data:

$$z_1 = \alpha_{11}x_1 + \alpha_{12}x_2 + ... + \alpha_{1n}x_n = \sum_{k=1}^{n} \alpha_{1k}x_k \quad (7)$$

The other components are calculated analogously and the j^{th} principal component must not be correlated to the previous components Jolliffe1986:

$$z_j = \alpha_{j1}x_j + \alpha_{j2}x_j + ... + \alpha_{jn}x_n = \sum_{k=1}^{n} \alpha_{jk}x_k \quad (8)$$

4.4 Self-Organised Direction Aware Data Partitioning Algorithm(SODA)

In order to express this method, we must consider data space R^m and assume a data set as $\{x_1, x_2, x_3...\}$, where $x_i = [x_{i,1}, x_{i,2}, ..., x_{i,m}]^T \in R_m$ is a m dimensional vector, $i = 1, 2, 3, ...$; m is the dimensionality; subscript $i(i = 1, 2, 3, ...)$ indicate the time instances at which the i^{th} data sample arrives. Therefore, within the observed data set at the n^{th} time instance denoted by $\{x_1, x_2, ..., x_n\}$, we also consider the set of sorted unique values of data samples $\{u_1, u_2, ..., u_{n_u}\}$ $(u_i = [u_{i,1}, u_{i,2}, ..., u_{i,m}]^T \in R_m)$ from $\{x_1, x_2, ..., x_n\}$ and the corresponding normalized numbers of repeats $\{f_1, f_2, ..., f_n\}$, where $n_u(1 < n_u \leq n)$ is the number of unique data samples and $\sum_{i=1}^{n_u} f_i = 1$. The following derivations are conducted at the n^{th} time instance as a default unless there is a specific declaration SODA.

Distance/Dissimilarity Components in SODA The SODA approach, in this work, employs SODA:

i a magnitude component $d_M(x_i, x_j)$ based on the euclidean distance metric;
ii a angular $d_A(x_i, x_j)$ component based on the cosine similarity;

EDA Operators The recently introduced Empirical Data Analytics (EDA) SODA is an alternative methodology for machine learning which is entirely based on actual empirically observed data samples.

The EDA operators includeSODA:

i. **Cumulative Proximity(Gu et al., 2018):**
The cumulative proximity, π of $x_i(i = 1, 2, ..., n)$ is defined as:

$$\pi_n(x_i) = \sum_{j=1}^{n} d^2(x_i, x_j) \quad (9)$$

where $d(x_i, x_j)$ denotes the distance/dissimilarity between x_i and x_j.

ii. **Local Density (Gu et al., 2018):**
Local density D is defined as the inverse of the normalized cumulative proximity and it directly indicates the main pattern of the observed data. The local density, D of i $x_i(i = 1, 2, ..., n; n_u > 1)$ is defined as follows:

$$D_n(x_i) = \frac{\sum_{j=1}^{n} \pi_n(x_j)}{2n\pi_n(x_i)} \quad (10)$$

In the proposed SODA data partitioning approach, since both components, the magnitude (metric) and the angular one are equally important, the local density of $x_i(i = 1, 2, ..., n; n_u > 1)$ is defined as the sum of the metric/Canberra-based local density $(D_n^M(x_i))$ and the angular-based local density $(D_n^A(x_i))$.

iii. **The Global Density (Gu et al., 2018):**
The global density is defined for unique data samples together with their corresponding numbers of repeats in the data set/stream. It has the ability of providing multi-modal distributions automatically without the need of user decisions, search/optimization procedures or clustering algorithms. The global density of a particular unique data sample, $u_i(i = 1, 2, ..., n_u; n_u > 1)$ is expressed as the product of its local density and its number of repeats considered as a weighting factor as follows:

$$D_n^G(u_i) = f_i D_n(u_i) \quad (11)$$

As we can see from the above equations, the main EDA operators: cumulative proximity (π), local density (D) and global density (D^G) can be updated recursively, which shows that the proposed SODA algorithm is suitable for online processing of streaming data.

SODA Algorithm for Data Partitioning The main steps of the SODA algorithm include: firstly, form a number of DA planes from the observed data samples using both, the magnitude-based and angular-based densities; secondly, identify focal points, using the granularity γ of the clustering results and relates to the Chebyshev inequality SODA, we used $\gamma = 7.0$ in this work; finally, use the focal points to partition the data space into data-clouds. The detailed procedure of the proposed SODA partitioning algorithm is presented by SODA.

4.5 Grouping algorithm

This algorithm gathers all data-clouds that contain data pertaining to the same group. The groups are adequate condition tool data ($Index = 0$), intermediate condition tool data ($Index = 1$) and inadequate condition tool data ($Index = 2$), as presented in Section 1. Accordingly, the grouping algorithm associates each data sample to a label that is used in the classification module (Fernandes et al., 2022). The output provided by SODA is a vector composed by the indexes that indicate from which data-cloud each data sample belongs. Taking the number of data samples into consideration for each data-cloud, the percentage of data relating to each group (0, 1 or 2) was determined. The Hard Voting method is used to create the target vector. Hence, the time series are labeled according to the majority group in each data cloud (Fernandes et al., 2022).

The main objectives of this algorithm are to reduce human interference and optimize the number of groups considered in the classification task. The grouping algorithm considers both the user and the SODA labels when performing the data categorization. Therefore it combines the optimal number of groups with the data-driven and non-parametric properties of the data clouds provided by SODA (Fernandes et al., 2022).

5. EXPERIMENTAL RESULTS

In this Section, all the algorithms were executed on a computer with an Intel Core i7-8565U processor with a clock frequency of 4.60 GHz and 12 GB of RAM. The data set used in this work consists of 785 time-series recorded with the acquisition system described in Section 3 using an acquisition rate of 24 kHz. Each time series is a collection of 3584 measurements of electric voltage and current from the 3 phases of the lathes' spindle motor. The data set comprises 552-time series related to the adequate condition tool, 80 related to intermediate condition tool and 153 related to the inadequate condition tool.

The train and test subsets were divided using k-fold cross-validation with 5 folds.

The TSFRESH algorithm, was used as the solution for features extraction/selection of this work. As presented in Section 4.2, features are extracted with different parameters. The feature selection strongly depends on the data patterns of the training subset. Hence the number of extracted features may differ along the cross-validation.

The feature selection resulted in 1642 relevant features using the union strategy and 655 relevant features using the intersection strategy for each time series. It is worth mentioning that, these strategies were presented in Section 4.2. Afterward, the PCA method was applied to the TSFRESH output data aiming to reduce its dimensionality. Owing to the fact that using more components would not increase significantly the maintained information the first 4 PCs were kept.

5.1 Classification

The SODA (Gu et al., 2018) divided the data into 8 data clouds. The grouping algorithm, exposed in Section 2, divided these clouds into groups as follows: adequate condition tools' clouds, intermediate condition tools' clouds, inadequate condition tools' clouds. The data was labeled according to this division. The train and test subsets were divided using k-fold cross-validation with 5 folds. For comparison purposes, the classification was executed with union and intersection on the features selection provided by TSFRESH. The classification results for each case are presented in Table 1.

Considering Table 1, for the majority of classifiers, the features union strategy results in higher accuracy than the intersection strategy. On the other hand, the union strategy doubles the algorithm's execution time. Consequently, the accuracy gain does not justify the use of features union.

Considering the classification results, Nearest Neighbors, Radial-basis function kernel Gaussian Process, Decision Tree and Random Forest delivered similar performances in term of balanced accuracy and all classifiers, when using the selected features intersection, exhibited a similar performance in terms of elapsed time. However, balancing both accuracy and elapsed time, the Nearest Neighbor classifier using selected features intersection exhibited the best performance. Therefore, to exemplify the final results this classifier was selected and Figure 6 presents its confusion matrix.

The classifiers implemented in this work were based on scikit-learn (Pedregosa et al., 2011), an open-source machine learning library in python. Even though other configurations, for the classifiers, were experimented, the maximum accuracy was achieved with the configurations presented in the example that follows `https://scikit-learn.org/stable/auto_examples/classification/plot_classifier_comparison.html`, except from the Decision Tree, the Random Forest and the MLP methods.

In the Decision tree and in the Random Forest methods, the maximum depth of the tree was not defined, therefore the nodes are expanded until all leaves are pure or until all leaves contain less than 2 samples. In the MLP the maximum number of iterations was set to 200. In the Linear SVM and in the Radial-basis function kernel SVM, the multi-class support is handled according to a one-vs-one scheme. In the Radial-basis function kernel Gaussian

Table 1. Classification Balanced Accuracy

Classifier	Balanced Accuracy[%]		Time [min]	
	Union	Intersection	Union	Intersection
Nearest Neighbors	86.18 ± 2.02	86.50 ± 3.40	7:22.220 ± 0:13.210	3:28.626 ± 0:23.626
Linear SVM	53.03 ± 6.20	51.95 ± 3.49	7:22.215 ± 0:13.211	3:28.620 ± 0:23.625
Radial-basis function kernel SVM	53.99 ± 5.26	69.48 ± 4.30	7:22.216 ± 0:13.211	3:28.621 ± 0:23.625
Radial-basis function kernel Gaussian Process	86.27 ± 1.83	83.47 ± 3.57	7:22.238 ± 0:13.211	3:28.645 ± 0:23.628
Decision Tree	83.02 ± 4.37	80.42 ± 2.46	7:22.214 ± 0:13.211	3:28.619 ± 0:23.625
Random Forest	83.73 ± 2.46	83.27 ± 2.06	7:22.224 ± 0:13.211	3:28.630 ± 0:23.625
MLP neural network	74.94 ± 3.78	74.47 ± 6.25	7:22.214 ± 0:13.211	3:28.619 ± 0:23.625
AdaBoost	74.41 ± 3.86	78.18 ± 6.60	7:22.222 ± 0:13.211	3:28.627 ± 0:23.625
Gaussian Naive Bayes	66.58 ± 2.76	64.16 ± 5.68	7:22.214 ± 0:13.211	3:28.619 ± 0:23.625
Quadratic Discriminant Analysis	69.76 ± 2.79	63.92 ± 5.46	7:22.214 ± 0:13.211	3:28.619 ± 0:23.625

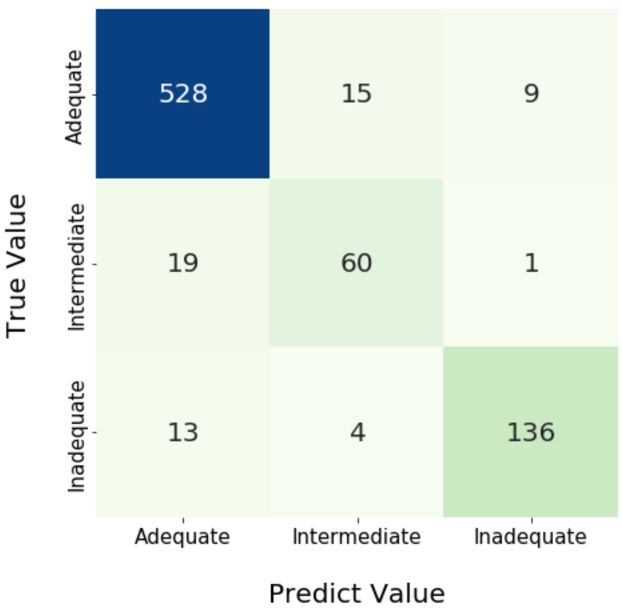


Figure 6. Confusion Matrix for Nearest Neighbors Classifier using selected features intersection.

Process, the multi-class support is handled according to a one-vs-rest scheme.

6. CONCLUSION

This paper proposed an approach to classify lathe's cutting tools based on TSFRESH, SODA and Machine Learning techniques on a multi-class scenario. Considering the classification results, Nearest Neighbors, Radial-basis function kernel Gaussian Process, Decision Tree and Random Forest delivered a performance above 80% of balanced accuracy. Since this work deals with an imbalanced multi-class classification and the balanced accuracy is the average of Recall obtained on each class, these results indicate that our model misclassifies only a small amount of samples.

The proposed model can identify the patterns that distinguish the cutting tool operations as adequate, intermediate, or in an inadequate condition, achieving satisfactory performances in all cases. Therefore, the proposed model allows preventing faulty pieces fabrication, waste of tools, and fault occurrences. Using this model, in a condition-based maintenance strategy, it is possible to increase the machining process reliability, quality, and availability, and reduce economic losses.

Furthermore, SODA algorithm provides a more accurate similarity recognition between the data than traditional clustering/partitioning methods, due to the fact that it considers both spatial and angular divergence. Moreover, it demonstrates an outstanding performance when applied to large-scale and high-dimensional situations without user-dependent parameters. Thus, decreasing the human interference in the proposed model application and granting a high computational efficiency, which supports the machine learning techniques in the classification task.

As an indication for future works, we intend to apply the SODA in the online processing of streaming data related to the lathe's motor time series analysis. Additionally, we intend to implement a different approach in the classification stage, such as fuzzy systems, aiming to generate an interpretable model capable to deal with uncertainties in the measure data. Also, the development of a low-cost embedded system capable of acquiring the data and classifying the tool condition in real-time.

ACKNOWLEDGMENTS

The authors would like to thank SESI/SENAI Juiz de Fora UI Luiz Adelar Scheuer, for the collaboration and hospitality in its facility during the data acquisition process. Also, the authors would like to acknowledge the Coordination for the Improvement of Higher Education Personnel - Brazil (CAPES) - Financing Code 001, National Council for Scientific and Technological Development - CNPq (process 433389 / 2018-4), Minas Gerais State Research Support Foundation - FAPEMIG (APQ-02922-18) and Federal University of Juiz de Fora - UFJF for financial support.

REFERENCES

Christ, M., Braun, N., Neuffer, J., and Kempa-Liehr, A.W. (2018). Time series feature extraction on basis of scalable hypothesis tests (tsfresh – a python package). *Neurocomputing*, 307, 72–77. doi:https://doi.org/10.1016/j.neucom.2018.03.067.

Fernandes, T.E. and de Aguiar, E.P. (2021). A new model to prevent failures in gas turbine engines based on tsfresh, self-organized direction aware data partitioning algorithm and machine learning techniques. *Journal of the Brazilian Society of Mechanical Sciences and Engineering*, 43(5). doi:https://doi.org/10.1007/s40430-021-02956-z.

Fernandes, T.E., Ferreira, M.A., de Miranda, G.P., Dutra, A.F., Antunes, M.P., da Silva, M.V., and Aguiar, E.P.d.

(2022). Classification of lathe's cutting tool wear based on an autonomous machine learning model. *Journal of Control, Automation and Electrical Systems*, 33(1), 167–182. doi:https://doi.org/10.1007/s40313-021-00819-5.

Gardner Business Media, I. (2021). 2020 world machine tool survey shows machine tool downturn. `https://www.gardnerweb.com/blog/post/2020-world-machine-tool-survey-shows-machine-tool-downturn`, Accessed: 2022-02-27.

Gu, X., Angelov, P.P., Kangin, D., and Principe, J.C. (2018). Self-organised direction aware data partitioning algorithm. *Information Sciences*, 423, 80–95. doi:https://doi.org/10.1016/j.ins.2017.09.025.

Lee, Y.K., Mavris, D.N., Volovoi, V.V., Yuan, M., and Fisher, T. (2010). A fault diagnosis method for industrial gas turbines using bayesian data analysis. *Journal of Engineering for Gas Turbines and Power*, 132(4). doi:https://doi.org/10.1115/1.3204508.

Li, X., Djordjevich, A., and Venuvinod, P.K. (2000). Current-sensor-based feed cutting force intelligent estimation and tool wear condition monitoring. *IEEE Transactions on Industrial Electronics*, 47(3), 697–702. doi:https://doi.org/10.1109/41.847910.

Pedregosa, F., Varoquaux, G., Gramfort, A., Michel, V., Thirion, B., Grisel, O., Blondel, M., Prettenhofer, P., Weiss, R., Dubourg, V., J., Vanderplas, Passos, A., Cournapeau, D., Brucher, M., Perrot, M., and Duchesnay, E. (2011). Scikit-learn: Machine learning in python. *Journal of Machine Learning Research*, 12(85), 2825–2830. URL `http://jmlr.org/papers/v12/pedregosa11a.html`.

Pinto, A.C.V., Fernandes, T.E., Silva, P.C.L., Guimarães, F.G., Wagner, C., and de Aguiar, E.P. (2022). Interval type-2 fuzzy set based time series forecasting using a data-driven partitioning approach. *Evolving Systems*. doi:https://doi.org/10.1007/s12530-022-09452-2.

Shao, H., Wang, H.L., and Zhao, X. (2004). A cutting power model for tool wear monitoring in milling. *International Journal of Machine Tools and Manufacture*, 44(14), 1503–1509. doi:https://doi.org/10.1016/j.ijmachtools.2004.05.003.

Steven Kline, J. (2021). Pandemic continues machine tool industry downturn, but u.s. now a top 4 producer. `https://www.mmsonline.com/articles/pandemic-continues-machine-tool-industry-downturn-but-us-now-a-top-3-producer`, Accessed: 2022-02-27.

Wilcox, R. (2005). Kolmogorov–smirnov test. *Encyclopedia of biostatistics*. doi:https://doi.org/10.1002/0470011815.b2a15064.